Product Information

JobScheduler Use Cases

Use Case Scenarios

Contents

- **Cross-Platform Scheduling**
  - Processing with Agents
  - Agentless Processing

- **File Watching**
  - Remote File Triggering
  - Remote File Processing

- **File Transfer**
  - Server-to-Server File Transfer
  - Push Files to Internet via Jump Host
  - Pull Files from Internet via Jump Host
  - Pull Files from DMZ in Near Real Time

- **Server Backup**
  - Automated Backup Handling

Cross-Platform Scheduling: Processing with Agents

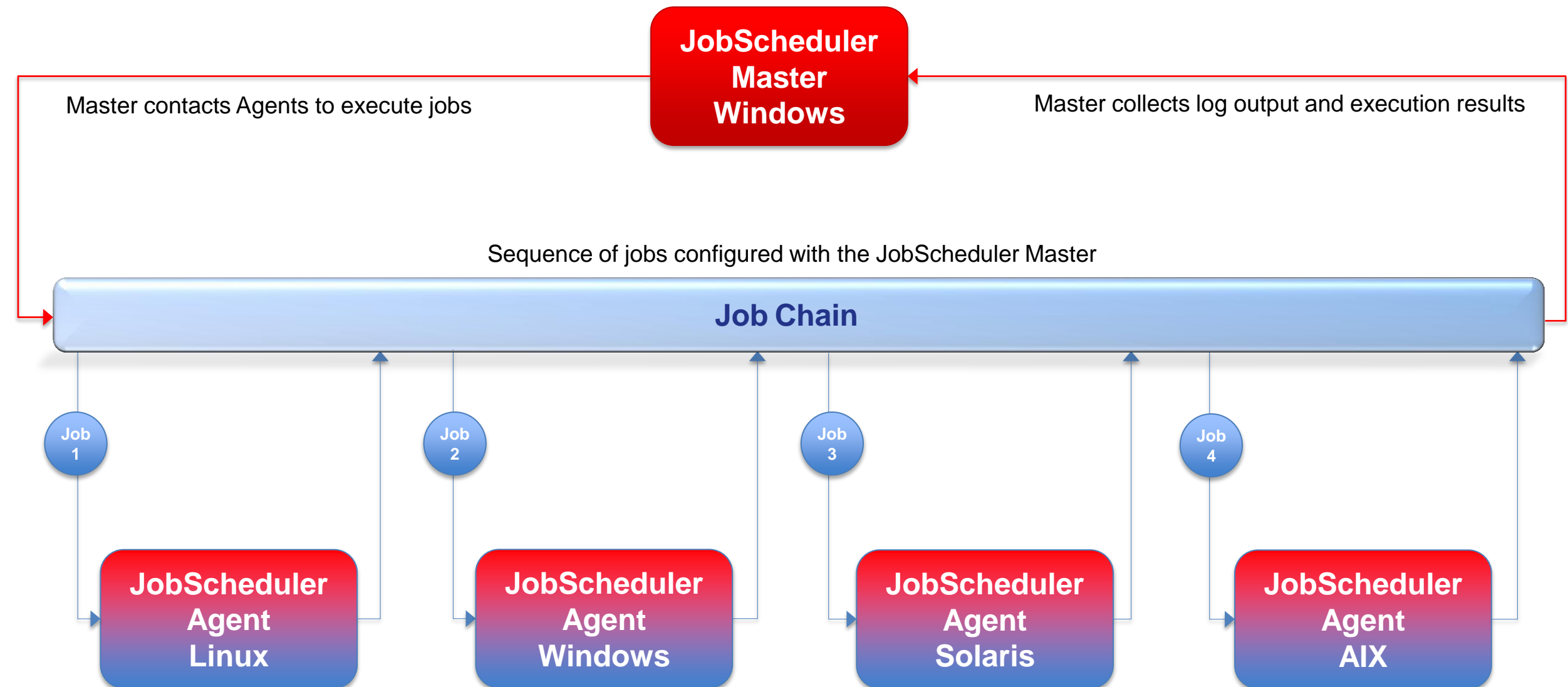Use Cases for typical scheduling tasks: Cross-Platform Scheduling

Use Case
- The customer operates servers with Windows, Linux, Solaris and AIX.
- Jobs with dependencies have to be executed on the different servers.

Solution
- A JobScheduler Master for Windows optionally executes jobs locally and orchestrates four Universal Agents.
- The job dependency is configured with the Master who contacts Agents to execute jobs.

Benefits
- JobScheduler covers all required platforms.
- Job dependencies for multi-platform execution can be configured.
- Single point of configuration and operation.
- Zero configuration for Universal Agents.

JobScheduler Master Windows

Master contacts Agents to execute jobs

Master collects log output and execution results

Sequence of jobs configured with the JobScheduler Master

Job Chain

Job 1

Job 2

Job 3

Job 4

JobScheduler Agent Linux

JobScheduler Agent Windows

JobScheduler Agent Solaris

JobScheduler Agent AIX

Cross-Platform Scheduling: Agentless Processing

Use Cases for typical scheduling tasks: Cross-Platform Scheduling
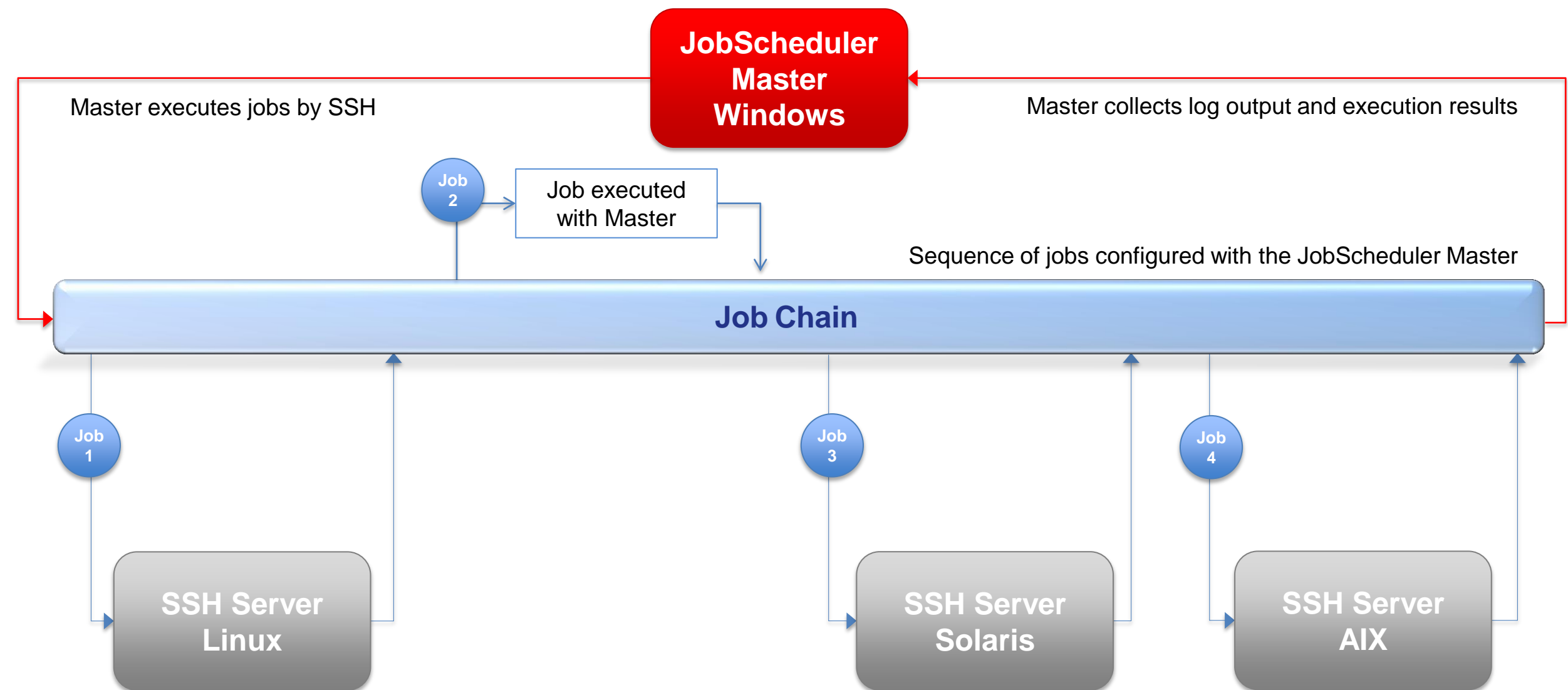
Use Case Scenario

Use Case
- The customer operates servers with Windows, Linux, Solaris and AIX.
- Jobs with dependencies have to be executed on the different servers.

Solution
- A JobScheduler Master for Windows executes jobs locally and orchestrates execution on three servers by SSH.
- The job dependency is configured with the Master who contacts Agents to execute jobs.

Benefits
- No software installation is required for job execution on existing SSH servers.
- Job dependencies for multi-platform execution can be configured.
- Single point of configuration and operation.

JobScheduler Master Windows

Master executes jobs by SSH

Master collects log output and execution results

Job 2 — Job executed with Master

Sequence of jobs configured with the JobScheduler Master

Job Chain

Job 1 — SSH Server Linux

Job 3 — SSH Server Solaris

Job 4 — SSH Server AIX

File Watching: Remote File Triggering

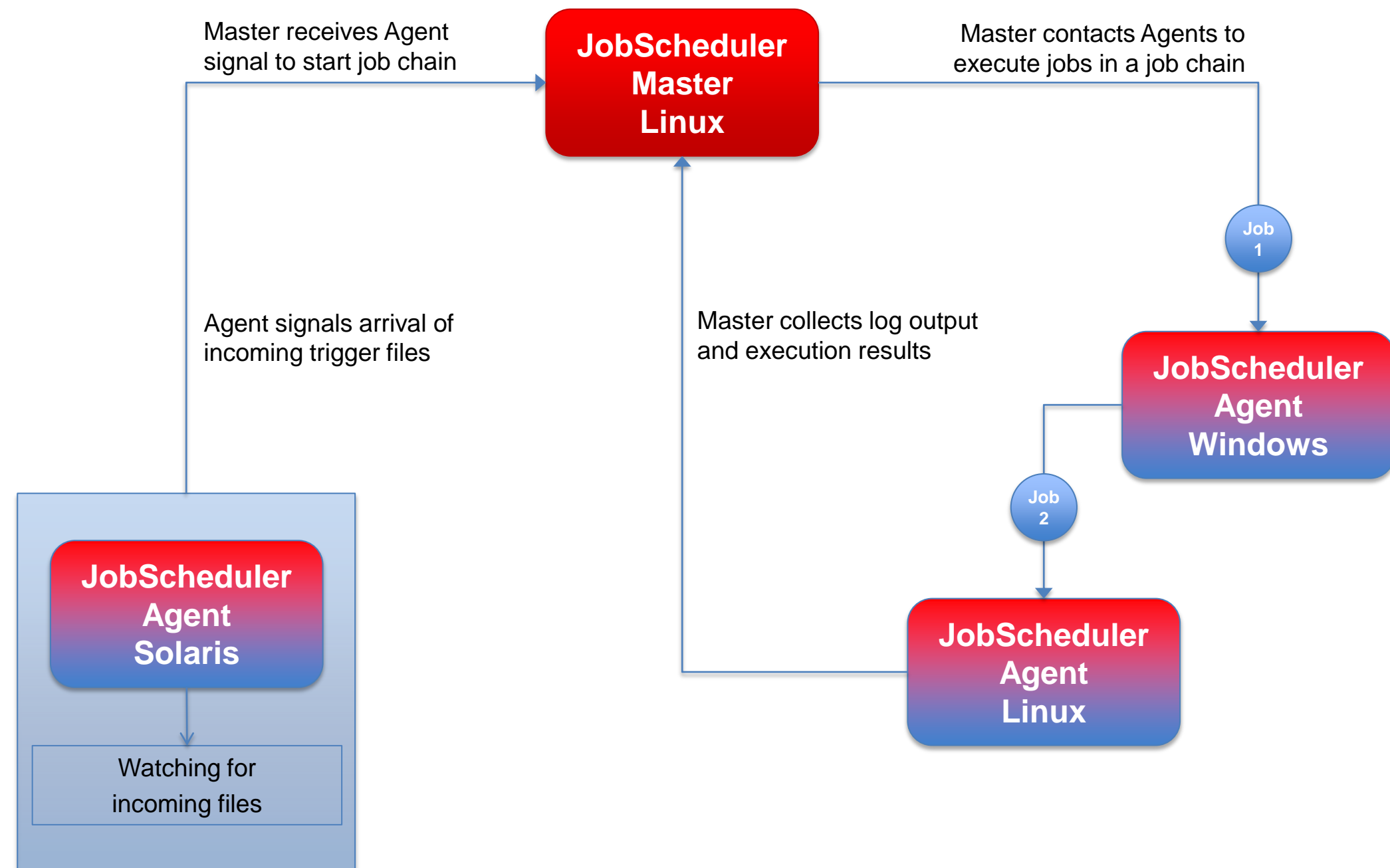Use Cases for typical scheduling tasks: File Watching

Use Case
- Files have to be watched on different platforms, e.g. on Solaris. Job starts shall be caused on Windows and Linux after arrival of a trigger file on Solaris.

Solution
- Universal Agent watches for incoming files and sends signal to the Master once a file arrives.
- The Master is configured to know which workflows are to follow.

Benefits
- Remote file watching with Master and Agents is platform independent.
- All configuration for the workflows are stored on the Master, the Universal Agents execute jobs with zero configuration.
- Remote file watching can be applied to any number of platforms and servers.

Master receives Agent signal to start job chain

JobScheduler Master Linux

Master contacts Agents to execute jobs in a job chain

Job 1

Agent signals arrival of incoming trigger files

Master collects log output and execution results

JobScheduler Agent Windows

JobScheduler Agent Solaris

Job 2

Watching for incoming files

JobScheduler Agent Linux

File Watching: Remote File Processing

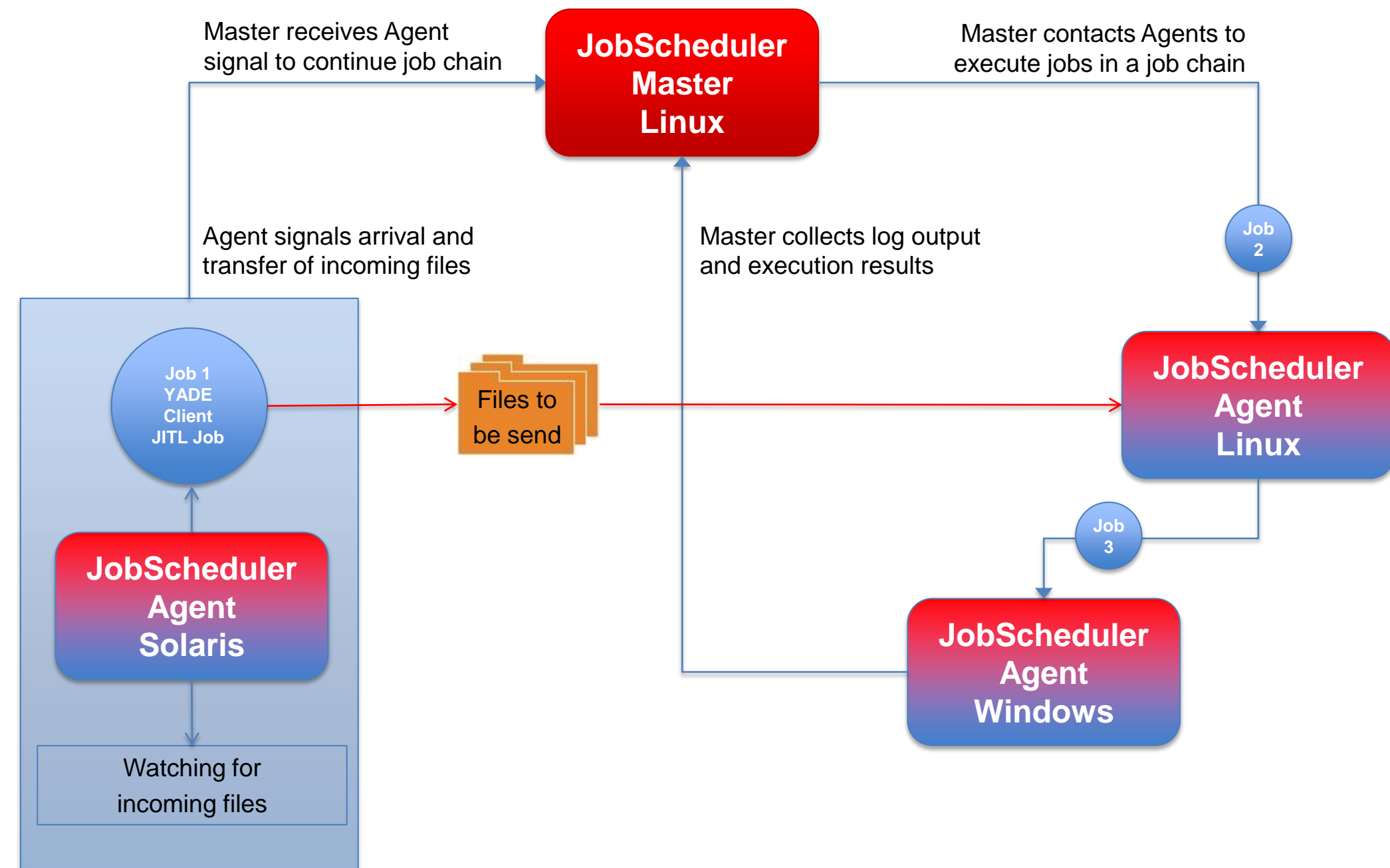Use Cases for typical scheduling tasks: File Watching

Use Case
- Files have to be watched on Solaris and sent to a Linux server. After transfer the files are processed on Linux and a successor job on Windows is executed.

Solution
- Universal Agent watches for incoming files and sends signal to the Master once a file arrives.
- The files are directly sent to a Linux server with a file transfer job on Solaris.
- The Master is configured to know which workflows are to follow.

Benefits
- Remote file watching with Master and Agents is platform independent.
- Configuration for the workflows are only required on the Master, the Universal Agents execute jobs with zero configuration.

Master receives Agent
signal to continue job chain

JobScheduler Master Linux

Master contacts Agents to
execute jobs in a job chain

Agent signals arrival and
transfer of incoming files

Master collects log output
and execution results

Job 2

Job 1 YADE Client JITL Job

Files to
be send

JobScheduler Agent Linux

Job 3

JobScheduler Agent Solaris

Watching for
incoming files

JobScheduler Agent Windows

File Transfer: Server-to-Server Transfer

Use Cases for typical scheduling tasks: Server-to-Server File Transfer
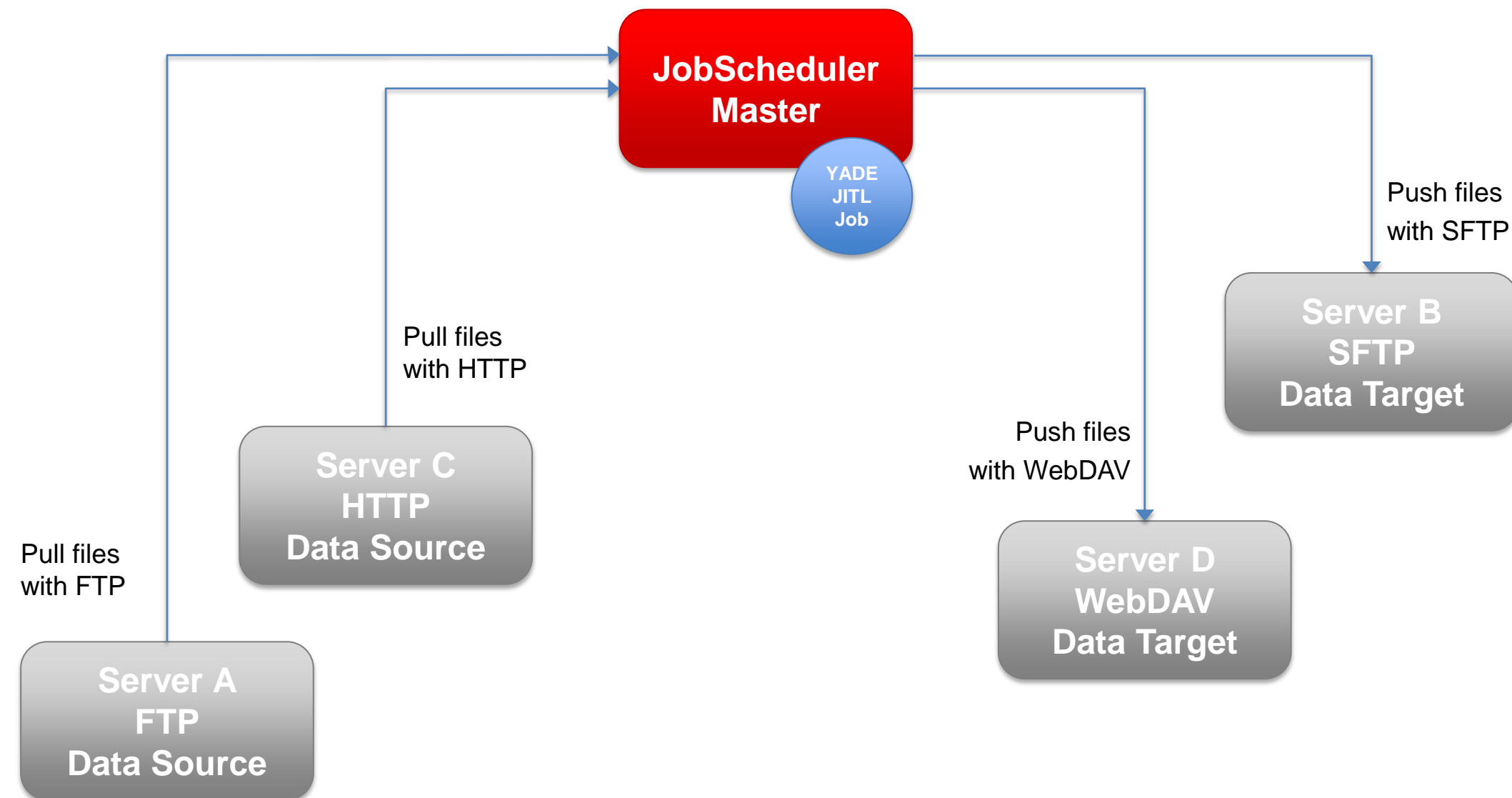
Use Case
- Files shall be transferred by using different transfer protocols for data sources and targets.

Solution
- A virtual file system approach is used by YADE. This allows to copy and move files between sources and targets with any given protocol.

Benefits
- Any combination of file transfer protocols is possible, e.g. FTP, FTPS, SFTP, HTTP, WebDAV.
- No touchdown of files on JobScheduler Master.
- Transfers are performed in memory, resulting in very economical use of resources.

JobScheduler Master

YADE JITL Job

Pull files with HTTP

Pull files with FTP

Push files with SFTP

Push files with WebDAV

Server C
HTTP
Data Source

Server A
FTP
Data Source

Server B
SFTP
Data Target

Server D
WebDAV
Data Target

File Transfer: Push Files to Internet via Jump Host

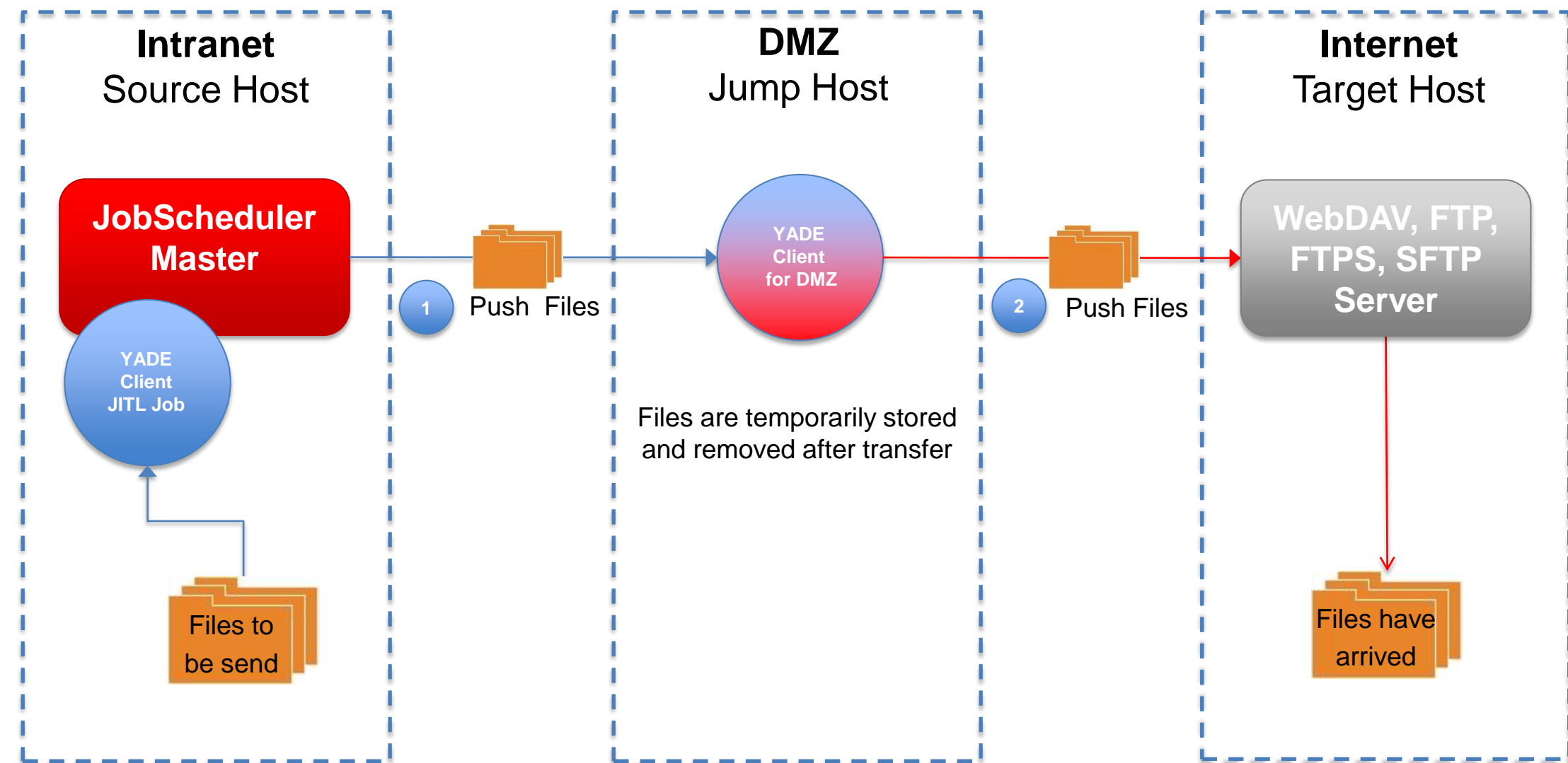Use Cases for typical scheduling tasks: File Transfer with Jump Host

Use Case

- Servers in the intranet have no internet access.
- Connection are only allowed from the intranet to the DMZ. From the DMZ transfers to the internet are allowed.

Solution

- YADE Client in the Intranet contacts YADE for DMZ to initiate a secure transfer of files from the intranet to the internet.
- YADE Client in intranet ① sends files to YADE in DMZ for temporary storing
- In a final step the YADE Client in DMZ ② pushs the files to the internet.

Benefits

- Files are stored temporarily in the DMZ and are removed after transfer.
- No credential data are stored in the DMZ.
- All configuration items are stored in the intranet.

Intranet
Source Host

DMZ
Jump Host

Internet
Target Host

JobScheduler Master

YADE Client JITL Job

① Push Files

YADE Client for DMZ

② Push Files

WebDAV, FTP, FTPS, SFTP Server

Files are temporarily stored and removed after transfer

Files to be send

Files have arrived

File Transfer: Pull Files from Internet via Jump Host

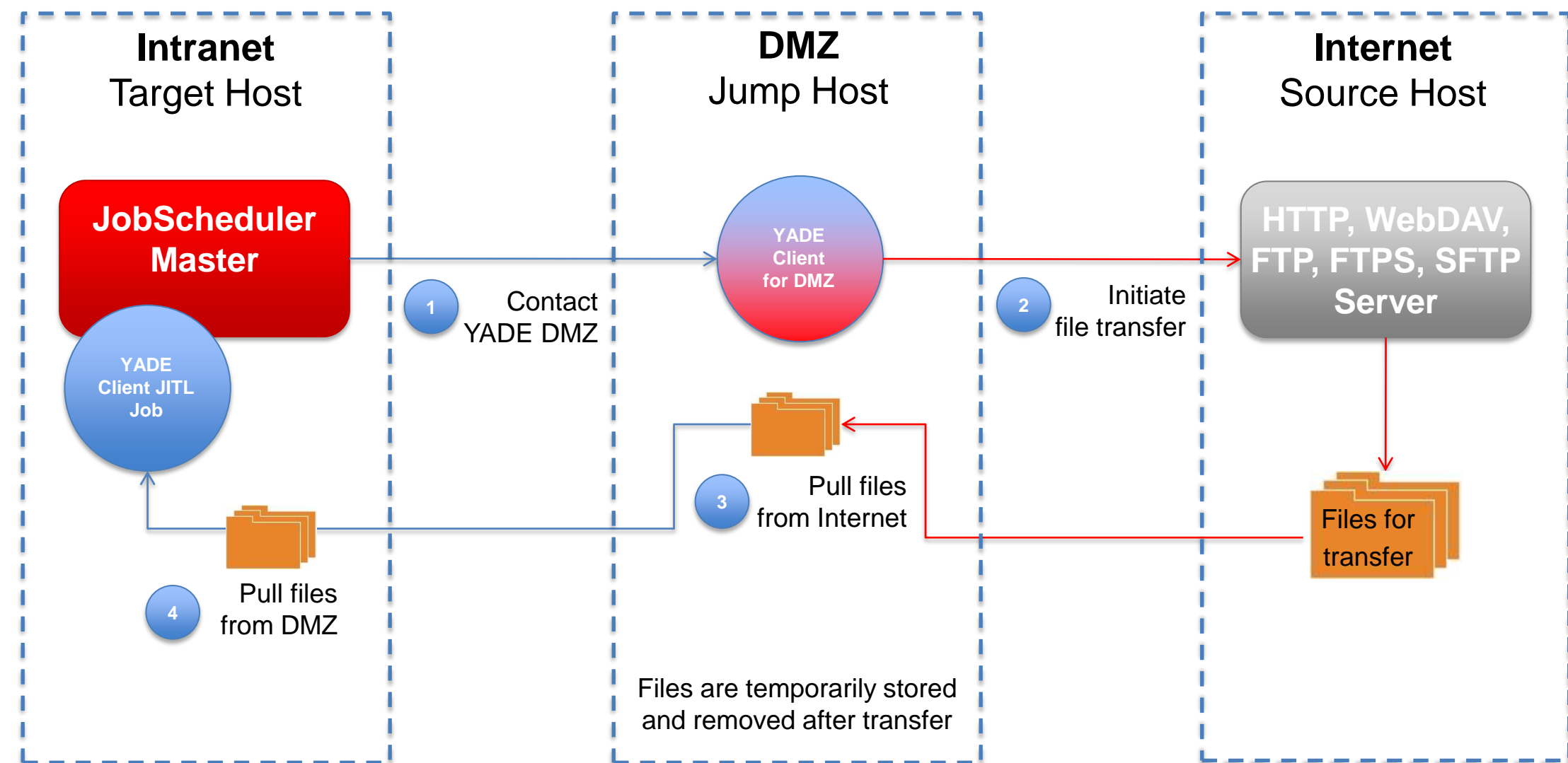Use Cases for typical scheduling tasks: File Transfer with Jump Host

Use Case

- Servers in the intranet have no internet access.
- Connection are only allowed from the intranet to the DMZ. From the DMZ transfers to the internet are allowed.

Solution

- The host in the DMZ acts as jump host for YADE.
- YADE Client in intranet ① contacts YADE in DMZ to ② start a transfer and to ③ pull files from the internet to the DMZ.
- YADE Client in intranet ④ pulls the files from the YADE in DMZ to the final destination in the intranet.

Benefits

- Files are stored temporarily in the DMZ and are removed after transfer.
- No credential data are stored in the DMZ.
- All configuration items are stored in the intranet.

Intranet
Target Host

DMZ
Jump Host

Internet
Source Host

JobScheduler Master

YADE Client JITL Job

YADE Client for DMZ

HTTP, WebDAV, FTP, FTPS, SFTP Server

① Contact YADE DMZ

② Initiate file transfer

③ Pull files from Internet

④ Pull files from DMZ

Files for transfer

Files are temporarily stored and removed after transfer

File Transfer: Pull Files from DMZ in Near Real Time

Use Cases for typical scheduling tasks: File Transfer with Jump Host

Use Case
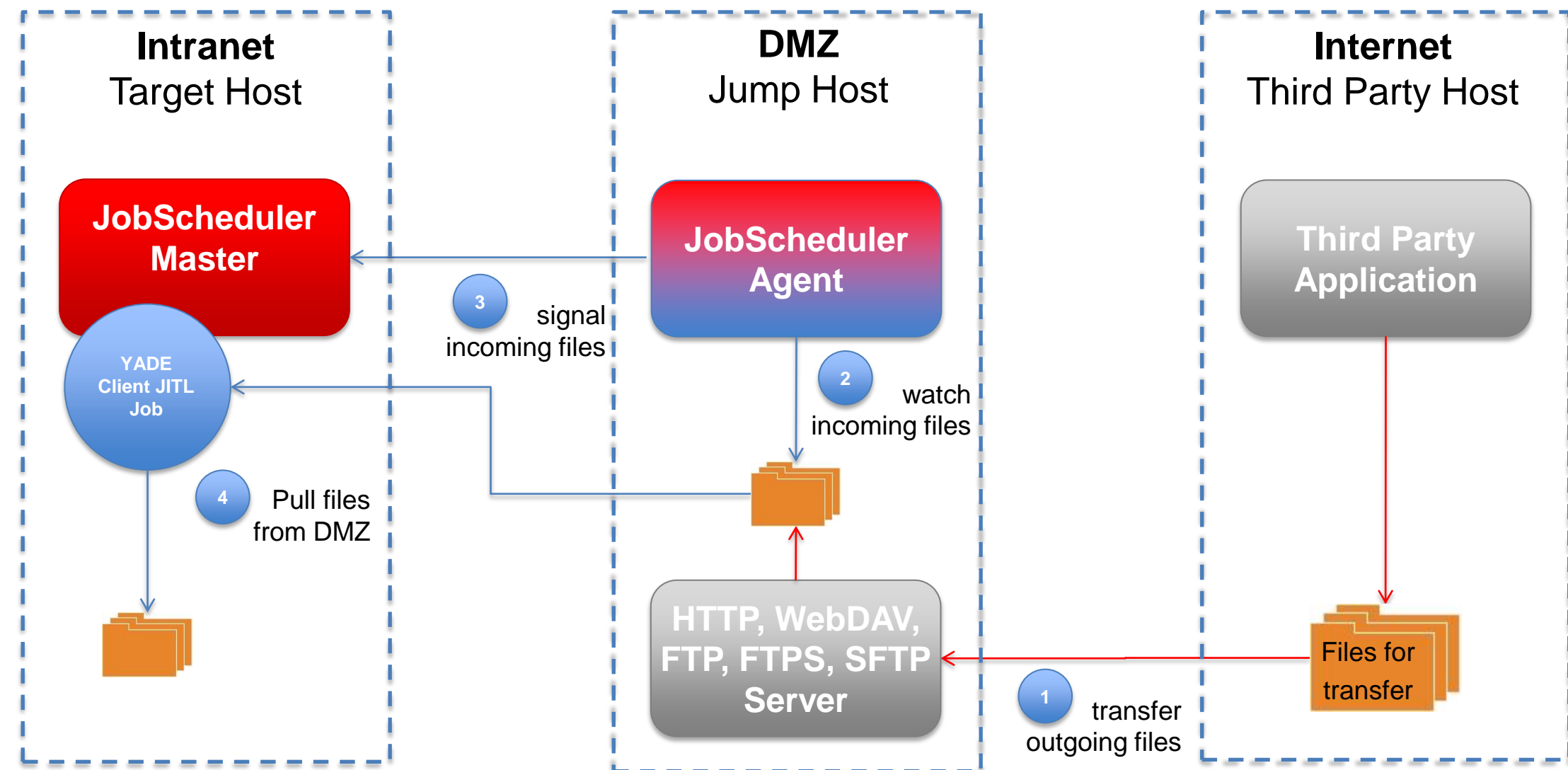
- Third Party pushs files to DMZ server.
- Incoming files should be transferred to the intranet as soon as possible in near real time.

Solution

- (1) Third Party transfers files to the DMZ server.
- (2) The Universal Agent runs on DMZ and monitors incoming files.
- (3) The Universal Agent signals to the Master the arrival of incoming files.
- (4) Master starts YADE job to transfer the files from DMZ to intranet.

Benefits

- Files are moved to intranet in near real time.
- Files are stored tempora-rily in the DMZ and are removed after transfer.
- All configuration items are stored in the intranet.

Intranet
Target Host

DMZ
Jump Host

Internet
Third Party Host

JobScheduler Master

YADE Client JITL Job

JobScheduler Agent

Third Party Application

3   signal incoming files

2   watch incoming files

4   Pull files from DMZ

HTTP, WebDAV, FTP, FTPS, SFTP Server

Files for transfer

1   transfer outgoing files

Server Backup: Automated Backup Handling

Use Cases for typical scheduling tasks: Server Backup
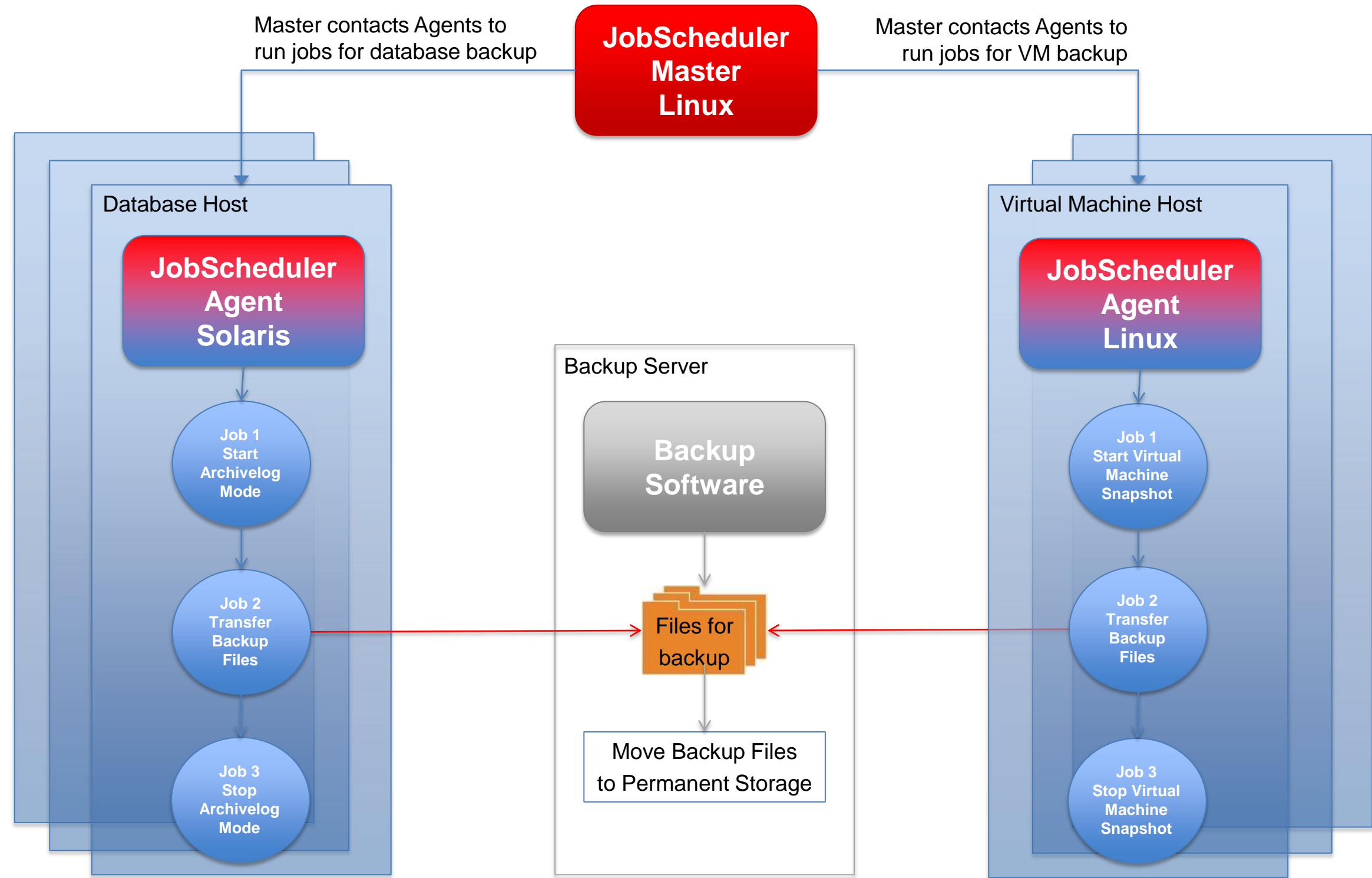
Use Case
- Backups shall be created for a number of database and virtual machine hosts.
- Backups shall be moved to a Backup Server for later permanént storage.

Solution
- Universal Agents are instructed to run job chains including pre-and post-processing for backup handling.
- The backups are directly sent to a Backup Server for later processing by some backup software.

Benefits
- Individual job chains for pre- and post-processing.
- Easy Agent deployment to a large number of servers.
- Configuration for the workflows are only required on the Master, the Universal Agents execute jobs with zero configuration.

Master contacts Agents to run jobs for database backup

JobScheduler Master Linux

Master contacts Agents to run jobs for VM backup

Database Host

JobScheduler Agent Solaris

Job 1 Start Archivelog Mode

Job 2 Transfer Backup Files

Job 3 Stop Archivelog Mode

Backup Server

Backup Software

Files for backup

Move Backup Files to Permanent Storage

Virtual Machine Host

JobScheduler Agent Linux

Job 1 Start Virtual Machine Snapshot

Job 2 Transfer Backup Files

Job 3 Stop Virtual Machine Snapshot

Questions?
Comments?
Feedback?

Software- und
Organisations-
Service GmbH

Giesebrechtstr. 15
D-10629 Berlin

info@sos-berlin.com
http://www.sos-berlin.com